Assignment #3

Jingye Wang

September 27, 2016

Gelman & Hill 11.4

The folder cd4 has CD4 percentages for a set of young children with HIV who were measured several times over a period of two years. The dataset also includes the ages of the children at each measurement.

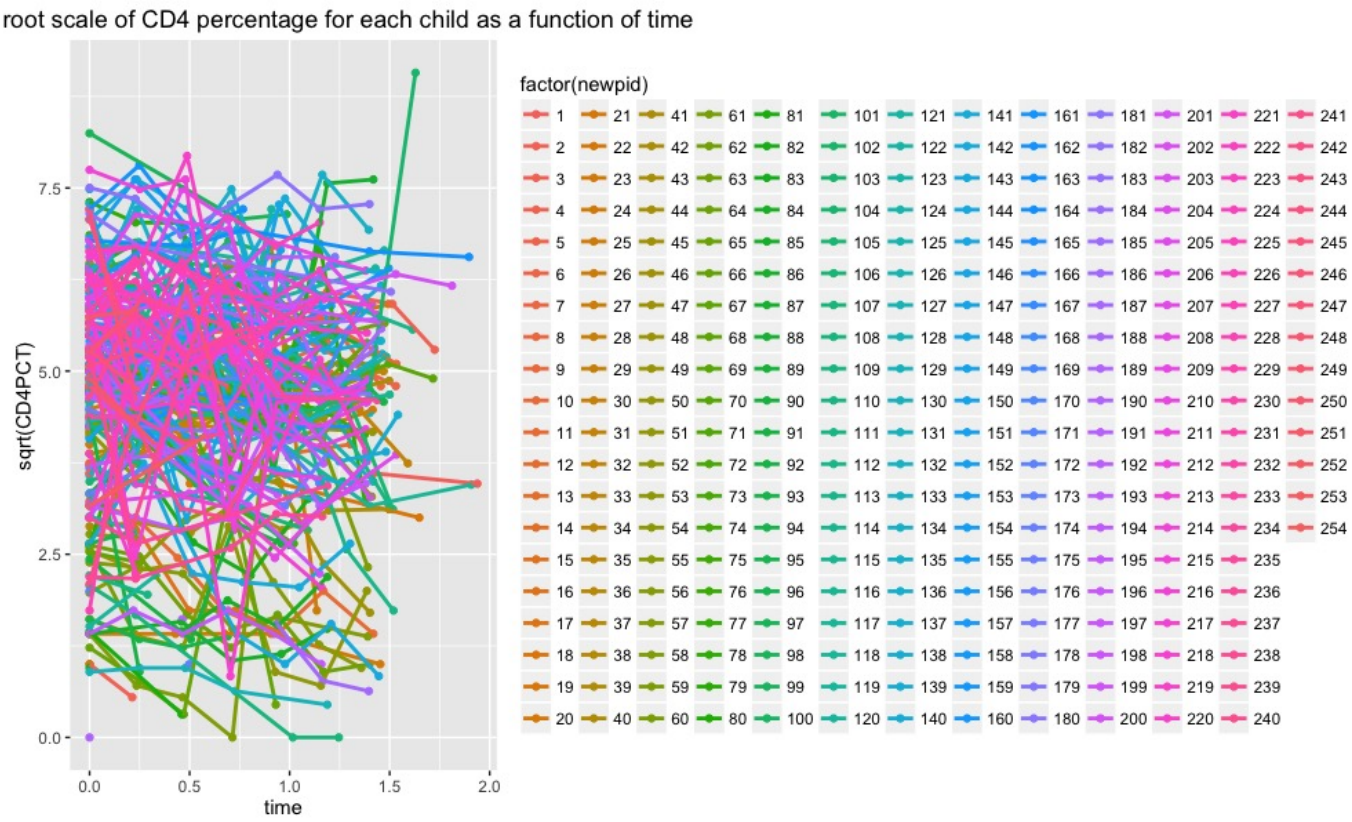
```
setwd("~/Dropbox/WUSTL third/Multilevel Modeling for Quantitative Research/assignmen
t/3")
.libPaths("/Library/Frameworks/R.framework/Versions/3.3/Resources/library")
# import data
hiv_data <- read.csv ("allvar.csv", header= T)
```

Part A

Graph the outcome (the CD4 percentage, on the square root scale) for each child as a function of time.

All lins in one plot

```
library(ggplot2)
hiv_data$time <- hiv_data$visage - hiv_data$baseage
ggplot(data=hiv_data,aes(x=time, y=sqrt (CD4PCT), color=factor(newpid))) + geom_point()
 +       geom_line(size = 1) + ggtitle('The square root scale of CD4 percentage for e
ach child as a function of time')
```

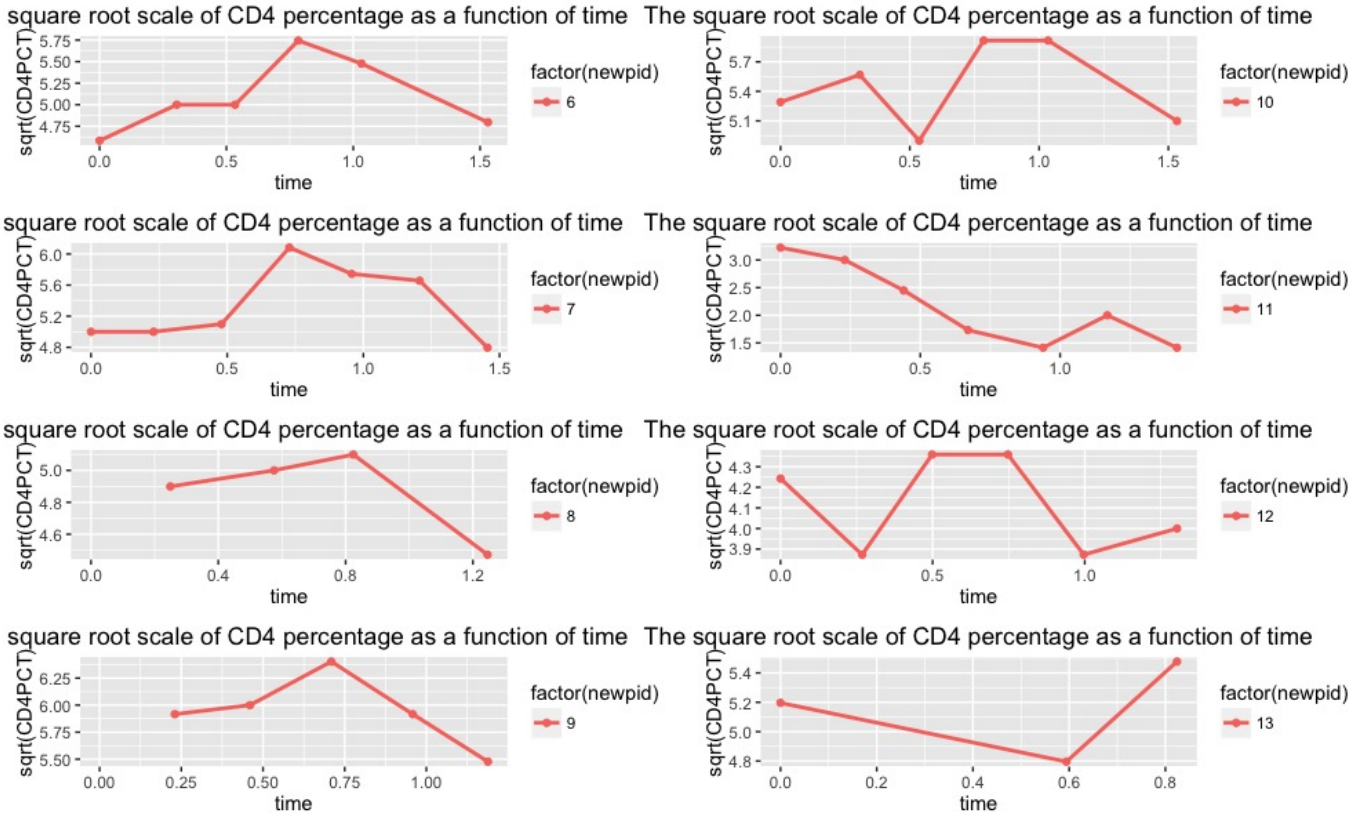


A sample of cases

Multiplot function

```
multiplot <- function(..., plotlist=NULL, file, cols=1, layout=NULL) {
  library(grid)

  # Make a list from the ... arguments and plotlist
  plots <- c(list(...), plotlist)

  numPlots = length(plots)

  # If layout is NULL, then use 'cols' to determine layout
  if (is.null(layout)) {
    # Make the panel
    # ncol: Number of columns of plots
    # nrow: Number of rows needed, calculated from # of cols
    layout <- matrix(seq(1, cols * ceiling(numPlots/cols)),
                     ncol = cols, nrow = ceiling(numPlots/cols))
  }

  if (numPlots==1) {
    print(plots[[1]])

  } else {
    # Set up the page
    grid.newpage()
    pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))

    # Make each plot, in the correct location
    for (i in 1:numPlots) {
      # Get the i,j matrix positions of the regions that contain this subplot
      matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))

      print(plots[[i]], vp = viewport(layout.pos.row = matchidx$row,
                                      layout.pos.col = matchidx$col))
    }
  }
}
```

```
plot_list = list()
for (i in 6:13){
p <- ggplot(data=hiv_data[hiv_data$newpid==i,],aes(x=time, y=sqrt (CD4PCT), color=fac
tor(newpid))) + geom_point() + geom_line(size = 1) + ggtitle('The square root scale o
f CD4 percentage as a function of time')
plot_list[[i]] = p
}
multiplot(plot_list[[6]],plot_list[[7]],plot_list[[8]],plot_list[[9]],plot_list[[10]],p
ot_list[[11]],plot_list[[12]],plot_list[[13]],cols=2)
```

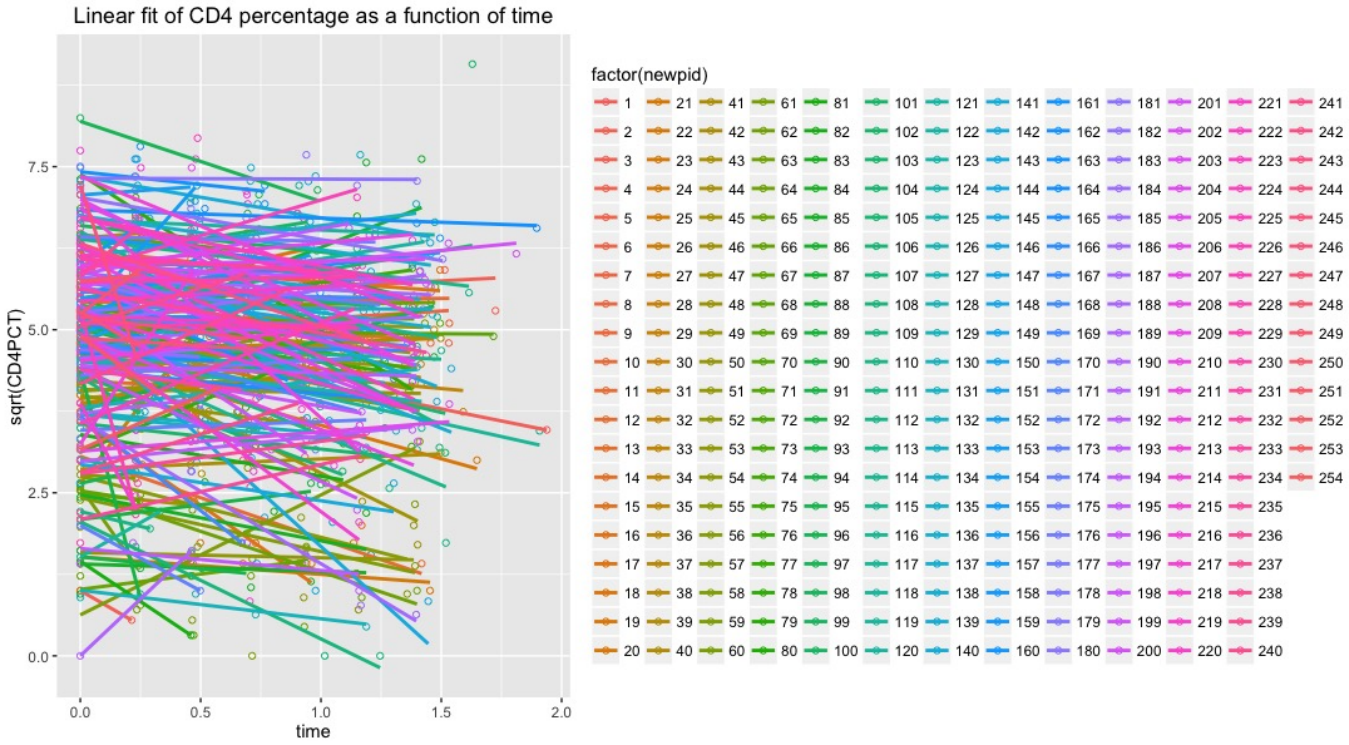


Part B

Each child's data has a time course that can be summarized by a linear fit. Estimate these lines and plot them for all the children.

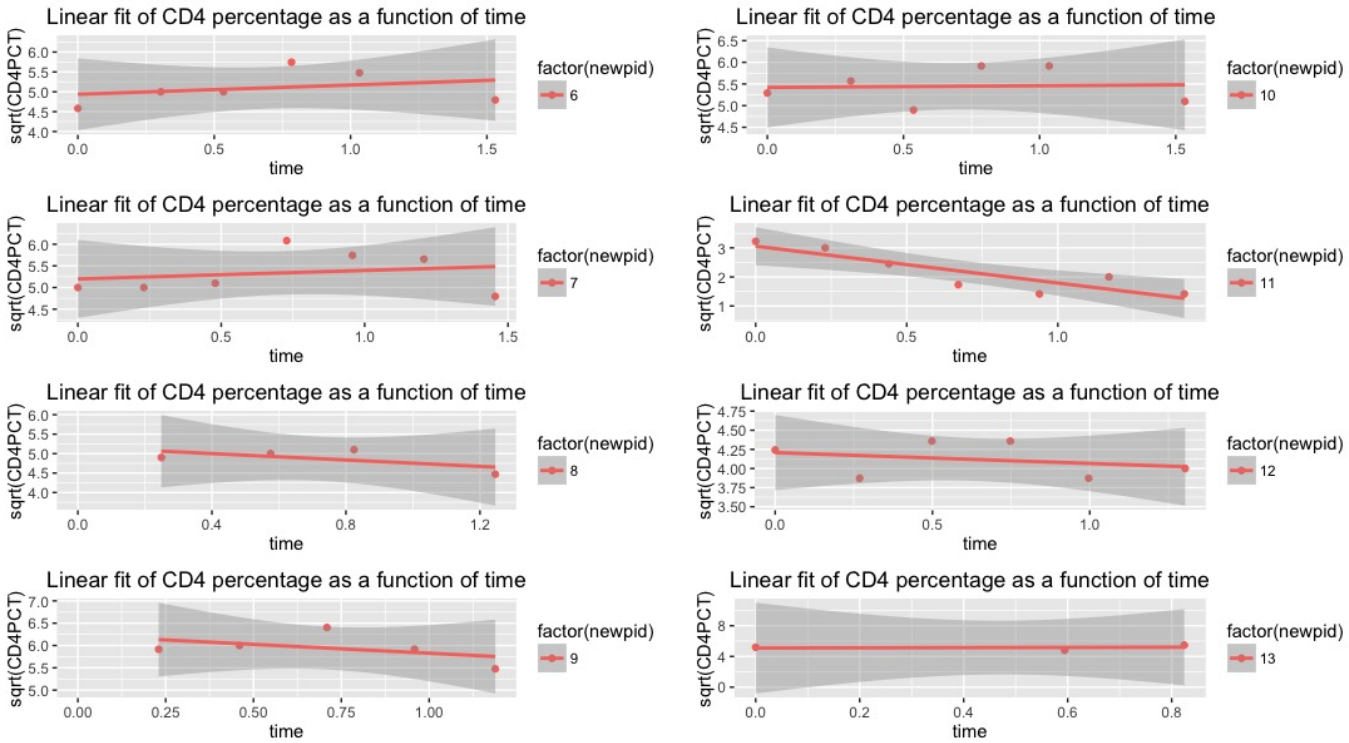
All lins in one plot

```
library(ggplot2)
ggplot(hiv_data, aes(x=time, y=sqrt (CD4PCT), color=factor(newpid))) + geom_point(sha
pe=1) + geom_smooth(method=lm, se=FALSE) + ggtitle('Linear fit of CD4 percentage as a
 function of time')
```



A sample of cases

```
plot_list = list()
for (i in 6:13){
p <- ggplot(data=hiv_data[hiv_data$newpid==i,],aes(x=time, y=sqrt (CD4PCT), color=fac
tor(newpid))) + geom_point() + geom_smooth(method=lm) + ggtitle('Linear fit of CD4 pe
rcentage as a function of time')
plot_list[[i]] = p
}
multiplot(plot_list[[6]],plot_list[[7]],plot_list[[8]],plot_list[[9]],plot_list[[10]],p
ot_list[[11]],plot_list[[12]],plot_list[[13]],cols=2)
```



Part C

Set up a model for the children's slopes and intercepts as a function of the treatment and age at baseline. Estimate this model using the two-step procedure–first estimate the intercept and slope separately for each child, then fit the between-child models using the point estimates from the first step.

Step one

lmList() can fit a list of lm models for different subgroups of the data

```
library(lme4)
model_2 <- lmList(sqrt(CD4PCT) ~ time|newpid, data = hiv_data)
coef_df <- data.frame(coef(model_2))
colnames(coef_df) <- c('Inter', 'Slope')
child_df <- data.frame(ID = hiv_data$newpid, treatment = hiv_data$treatmnt, baseage =
 hiv_data$baseage)
child_df <- unique(child_df)
child_new_df <- merge(child_df, coef_df, by.y = 'row.names', by.x = 'ID')
child_new_df$treatment <- as.factor(child_new_df$treatment)
```

Step two

```
# model fit for intercept
inter_fit <- lm(Inter ~ treatment + baseage, data = child_new_df)
summary(inter_fit)
```

```
Call:
lm(formula = Inter ~ treatment + baseage, data = child_new_df)

Residuals:
    Min      1Q  Median      3Q     Max
-5.0665 -0.7762  0.1892  1.0817  3.0391

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.11787    0.19048  26.868  < 2e-16 ***
treatment2   0.12364    0.18736   0.660  0.50992
baseage     -0.12100    0.04092  -2.957  0.00341 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.48 on 247 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared:  0.03577,   Adjusted R-squared:  0.02796
F-statistic: 4.581 on 2 and 247 DF,  p-value: 0.01112
```

```
# model fit for slope
slope_fit <- lm(Slope ~ treatment + baseage, data = child_new_df)
summary(slope_fit)
```

```
Call:
lm(formula = Slope ~ treatment + baseage, data = child_new_df)

Residuals:
     Min       1Q   Median       3Q      Max
-13.3917  -0.4547   0.2103   0.7651   6.0022

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.26568    0.27535  -0.965    0.336
treatment2  -0.13926    0.26936  -0.517    0.606
baseage     -0.04223    0.06016  -0.702    0.483

Residual standard error: 2.009 on 221 degrees of freedom
  (30 observations deleted due to missingness)
Multiple R-squared:  0.003496,  Adjusted R-squared:  -0.005523
F-statistic: 0.3876 on 2 and 221 DF,  p-value: 0.6791
```

Gelman & Hill 12.2

Continuing with the analysis of the CD4 data from Exercise 11.4

Part A

Write a model predicting CD4 percentage as a function of time with varying intercepts across children. Fit using lmer() and interpret the coefficient for time.

Model fit

```
library(lme4)
model_3 <- lmer(sqrt(CD4PCT) ~ time + (1 | newpid), data= hiv_data)
summary(model_3)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: sqrt(CD4PCT) ~ time + (1 | newpid)
   Data: hiv_data

REML criterion at convergence: 3140.8

Scaled residuals:
    Min      1Q  Median      3Q     Max
-4.7379 -0.4379  0.0024  0.4324  5.0017

Random effects:
 Groups   Name        Variance Std.Dev.
 newpid   (Intercept) 1.9569   1.3989
 Residual             0.5968   0.7725
Number of obs: 1072, groups:  newpid, 250

Fixed effects:
            Estimate Std. Error t value
(Intercept)  4.76341    0.09648   49.37
time        -0.36609    0.05399   -6.78

Correlation of Fixed Effects:
     (Intr)
time -0.278
```

Interpret the coefficient for time

In this model, each increase of 1% in time corresponds to a 0.37% predicted decrease in the square root scale of CD4 percentage.

Part B

Extend the model in (a) to include child-level predictors (that is, group-level predictors) for treatment and age at baseline. Fit using lmer() and interpret the coefficients on time, treatment, and age at baseline.

Model fit

```
model_4 <- lmer(sqrt(CD4PCT) ~ time + factor(treatmnt) + baseage + (1 | newpid),
data= hiv_data)
summary(model_4)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: sqrt(CD4PCT) ~ time + factor(treatmnt) + baseage + (1 | newpid)
   Data: hiv_data

REML criterion at convergence: 3137.2

Scaled residuals:
    Min      1Q  Median      3Q     Max
-4.7490 -0.4392  0.0097  0.4282  5.0141

Random effects:
 Groups   Name        Variance Std.Dev.
 newpid   (Intercept) 1.8897   1.3747
 Residual             0.5969   0.7726
Number of obs: 1072, groups:  newpid, 250

Fixed effects:
                  Estimate Std. Error t value
(Intercept)        5.08614    0.18793  27.064
time              -0.36216    0.05399  -6.708
factor(treatmnt)2  0.18008    0.18262   0.986
baseage           -0.11945    0.04000  -2.986

Correlation of Fixed Effects:
            (Intr) time   fct()2
time        -0.135
fctr(trtm)2 -0.462  0.010
baseage     -0.727 -0.017 -0.003
```

Interpret the coefficients on time, treatment, and age at baseline

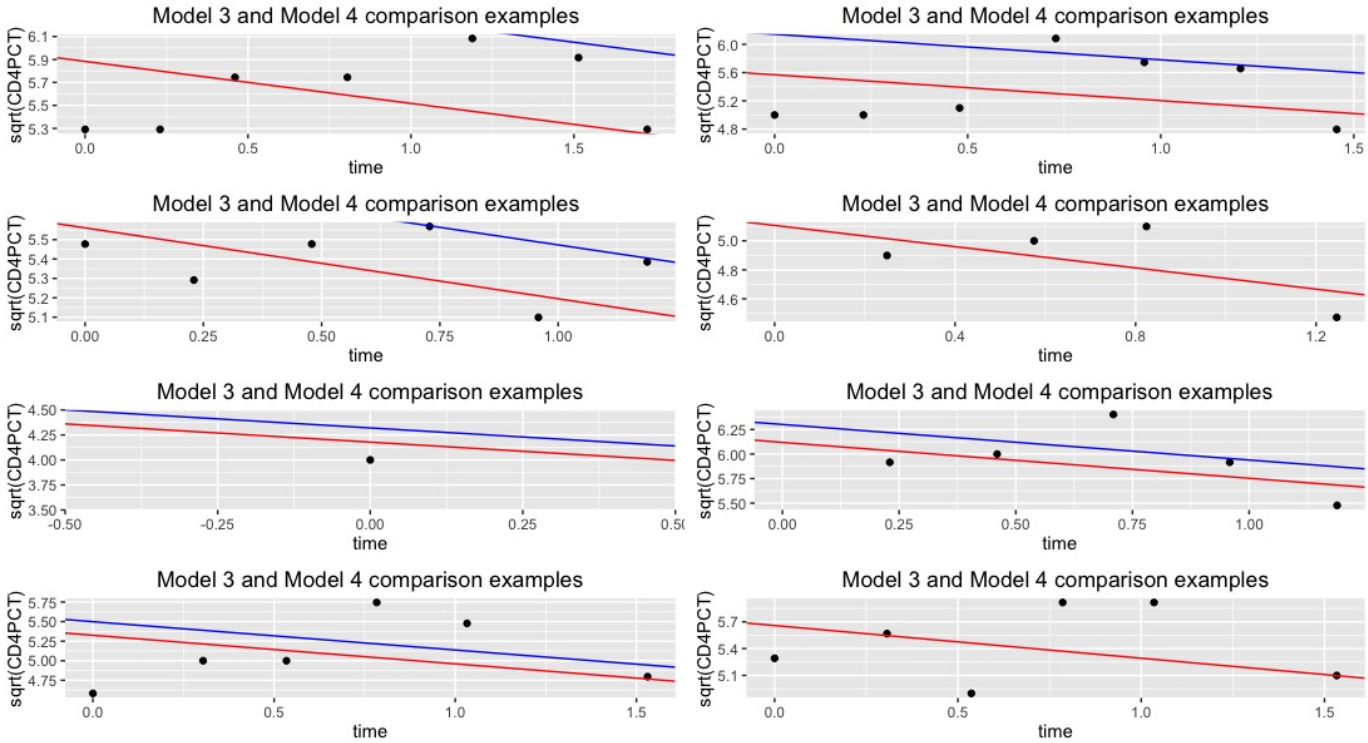
In this model, each increase of 1% in time corresponds to a 0.36% predicted decrease in the square root scale of CD4 percentage. Compared to treatment 1, treatment 2 corresponds to a 18% predicted increase in the square root scale of CD4 percentage. Each increase of 1% in baseline age corresponds to a 0.12% predicted decrease in the square root scale of CD4 percentage.

Part C

Investigate the change in partial pooling from (a) to (b) both graphically and numerically.

Plot

```
coef_df_3 <- data.frame(coef(model_3)$newpid)
colnames(coef_df_3) <- c('Inter', 'Slope')
coef_df_4 <- data.frame(coef(model_4)$newpid)
colnames(coef_df_4) <- c('Inter', 'Slope', 'Treatment', 'Baseage')

plot_list = list()
for (i in 3:10){
p <- ggplot(data = hiv_data[hiv_data$newpid==i, ], aes(time, sqrt(CD4PCT))) + geom_po
int() + geom_abline(slope = coef_df_3[i, 2], intercept = coef_df_3[i, 1], colour = 'r
ed') + geom_abline(slope = coef_df_4[i, 2], intercept = coef_df_4[i, 1], colour = 'bl
ue') + ggtitle('Model 3 and Model 4 comparison examples')
plot_list[[i]] = p
}
multiplot(plot_list[[3]],plot_list[[4]],plot_list[[5]],plot_list[[6]],plot_list[[7]],pl
t_list[[8]],plot_list[[9]],plot_list[[10]],cols=2)
```



numeric way

```
data <- cbind(coef_df_3, coef_df_4[,c(1,2)])
colnames(data) <- c('intercept_model_3', 'slope_model_3', 'intercept_model_4', 'slope
_model_4')
head(data)
```

```
  intercept_model_3 slope_model_3 intercept_model_4 slope_model_4
1          4.557250    -0.3660932          4.832595    -0.3621573
2          1.335566    -0.3660932          1.427542    -0.3621573
3          5.884129    -0.3660932          6.413092    -0.3621573
4          5.561130    -0.3660932          5.654863    -0.3621573
5          4.178397    -0.3660932          4.140021    -0.3621573
6          5.326751    -0.3660932          5.319323    -0.3621573
```

Numerically, it can be noted that the standard deviation of the group-level variable (intercept of patient ID) decreased from 1.3989 in the first model to 1.3747 in the second model. The standard deviation of residual does not change much (from 0.7725 to 0.7726), which means that the first model are more suitable to build a multi-level model. However, it can be noted that it is adequate for us to use the multilevel modeling for this dataset as the standard deviation for the group-level variable is big enough as compared with the residual in both models.

Gelman & Hill 12.5

Using the radon data, include county sample size as a group-level predictor and write the varying-intercept model. Fit this model using lmer().

import data

```
library(arm)
srrs2 <- read.table("srrs2.dat", sep = ',', header = T, row.names = 1)
cty <- read.table("cty.dat", sep = ',', header = T)
```

make a gropu-level dataset

```
srrs2$count <- unsplit(lapply(split(srrs2, srrs2[c("county")]), nrow), srrs2[c("count
y")])
srrs2.2<- as.data.frame(srrs2[!duplicated(srrs2$county), ])
```

Step 1: get county index variable

```
county.name <- as.vector(srrs2$county)
uniq <- unique(county.name)
J <- length(uniq)
county <- rep(NA, J)
for (i in 1:J) county[county.name==uniq[i]] <- i
```

Step 2: define n and y

```
radon <- srrs2$activity
countyvalue <- srrs2$county
n <- length(radon)
y <- radon

sample.size <- as.vector( table (county))
sample.size.jittered <- sample.size*exp(runif (J, -.1, .1))
ybarbar = mean(y)
pt.mns = tapply(y,county,mean)
pt.vars = tapply(y,county,var)
pt.sds = mean(sqrt(pt.vars[!is.na(pt.vars)]))/sqrt(sample.size)
```

Step 3: make connections between models

```
srrs2.county<- srrs2$county
srrs22.county<- srrs2.2$county
srrs.rows<-match (unique(srrs2.county),srrs22.county)
count<- srrs2.2[srrs.rows,"count"]
count.full <- count[county]
```

Step 4: run the model

```
model_5 <- lmer (radon ~  count.full  + (1 | county), srrs2)
summary(model_5)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: radon ~ count.full + (1 | county)
   Data: srrs2

REML criterion at convergence: 91965.3

Scaled residuals:
    Min      1Q  Median      3Q     Max
-2.8286 -0.2929 -0.1385  0.0528 29.1680

Random effects:
 Groups   Name        Variance Std.Dev.
 county   (Intercept) 12.02    3.467
 Residual             75.06    8.664
Number of obs: 12777, groups:  county, 386

Fixed effects:
             Estimate Std. Error t value
(Intercept)  4.944802   0.257867  19.176
count.full  -0.003240   0.003083  -1.051

Correlation of Fixed Effects:
           (Intr)
count.full -0.532
```